

**RECORDING AND REPRODUCING DEVICE**

Patent Number: JP11143787  
Publication date: 1999-05-28  
Inventor(s): TSUKADA MINORU; NAKAMURA KAZUO; INOUE KIYOSHI; KOTANI HIROAKI; NOZOE ATSUSHI;  
SHIODA SHIGEMASA; KATAYAMA YUKARI  
Applicant(s):: HITACHI LTD  
Requested Patent: ☐ JP11143787  
Application Number: JP19970304064 19971106  
Priority Number (s):  
IPC Classification: G06F12/16 ; G06F11/10  
EC Classification:  
Equivalents:

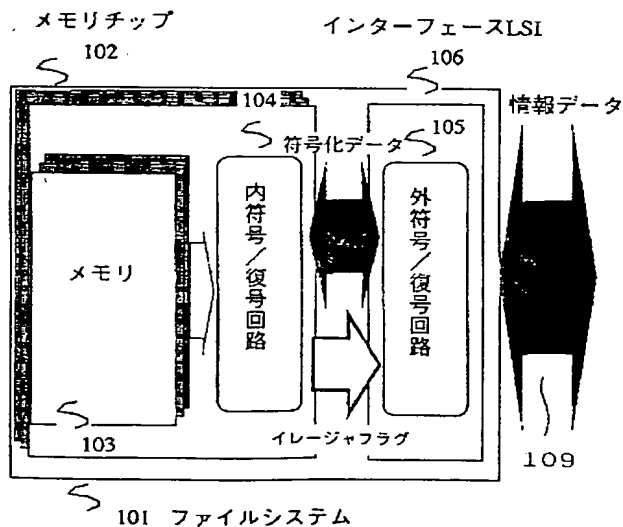
**Abstract**

**PROBLEM TO BE SOLVED:** To achieve the low error-correction-incapability of a file system without increasing the scale of an encoding/decoding circuit in a memory chip.

**SOLUTION:** An outside encoding/decoding circuit 105 outside a memory chip 102 generates an outside coded matrix by encoding each column of an information data matrix, and adding a redundant bit to each column. On the other hand, an inside encoding/decoding circuit 104 inside the memory chip 102 generates a product coded matrix by encoding the line direction of the outside coded matrix, and adding the redundant bit to each line, and stores it in a memory 103. When the memory chip 104 is used as a single body, the inside encoding/decoding circuit 104 generates the inside coded matrix by encoding the line direction of the information data matrix, and adding the redundant bit to each line, and stores it in the memory 103.

Data supplied from the esp@cenet database - I2

(11)特許出願公開番号



## 【特許請求の範囲】

【請求項 1】メモリを内蔵したメモリチップを用いた記録再生装置であって、

前記記憶再生装置は、前記メモリチップの外部に、記憶対象の情報データを誤り訂正符号化し第1の誤り訂正符号を生成する外符号/復号回路を備え、

前記メモリチップは、外符号/復号回路が生成した第1の誤り訂正符号を、さらに、誤り訂正符号化し第2の誤り訂正符号を生成し、前記メモリに記憶する内符号/復号回路を内蔵し、

前記内符号/復号回路は、前記メモリより読み出した第2の誤り訂正符号を用いて誤り訂正を行い、前記第1の誤り訂正符号を復号し、

前記外符号/復号回路は、前記内符号/復号回路が復号した第1の誤り訂正符号を用いて誤り訂正を行い、前記情報データを復号することを特徴とする記憶再生装置。

【請求項 2】請求項 1 記載の記憶再生装置であって、前記内符号/復号回路は、誤り訂正を行えなかった誤りの位置を示す位置情報を出力する手段を有し、前記外符号/復号回路は、前記内符号/復号回路が復号した第1の誤り訂正符号に加えて、前記内符号/復号回路から出力された位置情報を用いた誤り訂正を行うことを特徴とする記憶再生装置。

【請求項 3】請求項 1 記載の記憶再生装置であって、前記内符号/復号回路は、前記第2の誤り訂正符号の訂正能力に応じ、誤り訂正または誤り検出を行った誤りの位置を示す位置情報を出力する手段を有し、前記外符号/復号回路は、前記内符号/復号回路が復号した第1の誤り訂正符号に加えて、前記内符号/復号回路から出力された位置情報を用いた誤り訂正を行うことを特徴とする記憶再生装置。

【請求項 4】請求項 1 または 2 記載の記憶再生装置であって、

前記外符号/復号回路は、記憶対象の情報データをマトリクスとして扱い、当該マトリクスの各列/行を誤り訂正符号化して、複数列/行の第1の誤り訂正符号を生成し、

前記内符号/復号回路は、外符号/復号回路が生成した複数列/行の第1の誤り訂正符号を、行/列方向に並べたマトリクスの、各行/列を誤り訂正符号化して、複数行/列の第1の誤り訂正符号を生成し、当該複数行/列の第1の誤り訂正符号は、前記情報データの積符号を形成することを特徴とする記憶再生装置。

【請求項 5】請求項 1、2 または 3 記載の記憶再生装置であって、

前記第1の誤り訂正符号と第2の誤り訂正符号の少なくとも一方は、リードソロモン符号であることを特徴とする記憶再生装置。

【請求項 6】メモリを内蔵したメモリチップを用いた記録再生装置における誤り訂正方法であって、

前記記憶再生装置において、前記メモリチップの外部で、記憶対象の情報データを誤り訂正符号化し第1の誤り訂正符号を生成し、

前記メモリチップ内部において、外符号/復号回路が生成した第1の誤り訂正符号を、さらに、誤り訂正符号化し第2の誤り訂正符号を生成し、前記メモリに記憶し、前記メモリチップ内部において、前記メモリより読み出した第2の誤り訂正符号を用いて誤り訂正を行い、前記第1の誤り訂正符号を復号し、

10 前記記憶再生装置において、前記メモリチップの外部で、前記内符号/復号回路が復号した第1の誤り訂正符号を用いて誤り訂正を行い、前記情報データ復号することを特徴とする誤り訂正方法。

【請求項 7】メモリと、外部より供給されたメモリに書き込むべきデータを誤り訂正符号化し第1の誤り訂正符号を生成して前記メモリに記憶し、前記メモリより読み出した第1の誤り訂正符号を用いて誤り訂正を行い、前記データを復号し、外部に出力する第1の符号/復号回路とを内蔵したメモリチップの使用方法であって、

20 書き込み要求された情報データを誤り訂正符号化し第2の誤り訂正符号を生成し、第1の誤り訂正符号を用いて誤り訂正を行い、読み出し要求された情報データを復号する第2の符号/復号回路を備えた記憶装置において前記メモリチップを使用する場合には、前記に前記メモリに書き込むべきデータとして、第2の符号/復号回路において誤り訂正符号化し第2の誤り訂正符号を供給し、前記第1の符号/復号回路が復号し出力したデータを、前記誤り訂正を行うべき第1の誤り訂正符号として第2の符号復号回路に供給し、

30 前記第2の符号/復号回路を備えていない記憶再生装置において、記憶装置において前記メモリチップを使用する場合には、書き込み要求された情報データを、前記第1の符号復号回路に前記メモリに書き込むべきデータとして供給し、前記第1の符号/復号回路が復号し出力したデータを前記読み出し要求された情報データとすることを特徴とするメモリチップの使用方法。

【請求項 8】メモリと、外部より供給されたメモリに書き込むべきデータを誤り訂正符号化し第1の誤り訂正符号を生成して前記メモリに記憶し、前記メモリより読み出した第1の誤り訂正符号を用いて誤り訂正を行い、前記データを復号し、外部に出力する第1の符号/復号回路と、

前記内符号/復号回路が、誤り訂正を行えなかった誤りの位置を示す位置情報を外部に出力する手段とを内蔵したことを特徴とするメモリチップ。

【請求項 9】メモリと、

外部より供給されたメモリに書き込むべきデータを誤り訂正符号化し第1の誤り訂正符号を生成して前記メモリに記憶し、前記メモリより読み出した第1の誤り訂正符号を用いて誤り訂正を行い、前記データを復号し、外部

## 3

に出力する第1の符号/復号回路と、

前記内符号/復号回路が、第1の誤り訂正符号の訂正能力に応じ、誤り訂正または誤り検出を行った誤りの位置を示す位置情報を外部に出力する手段とを内蔵したことを特徴とするメモリチップ。

【請求項10】読み出されたデータに含まれる誤り位置を示す位置情報を出力するメモリチップの書き込み及び読み出しを制御するコントローラであって、記憶対象の情報データを誤り訂正符号化し誤り訂正符号を生成して前記メモリチップに書き込みデータとして供給し、前記メモリチップから読み出した第1の誤り訂正符号と、前記メモリチップから出力される前記位置情報とを用いて誤り訂正を行い、前記情報データを復号する符号/復号回路を備えたことを特徴とするコントローラ。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、メモリチップおよびメモリチップを利用した記録再生装置におけるエラー検出/訂正の技術に関するものである。

【0002】

【従来の技術】記録媒体からデータを読み出し転送する際には、さまざまな要因によりデータにエラーが発生する場合がある。これらの要因を大別すると、データを格納するメモリ素子に起因するエラーと、データ伝送時に伝送経路で発生するエラーがある。不揮発性半導体メモリ、特にフラッシュメモリにおいては、前者が優勢となり、特にメモリのリテンションエラーが問題となる。

【0003】以下、リテンションエラーの説明を行う。

【0004】まず、フラッシュメモリ素子の構造を図19に示す。

【0005】フラッシュメモリ素子において、データの書き込みは、浮遊ゲートへの電荷の注入あるいは、浮遊ゲートからの電荷の引き抜きでおこなう。そして、データの読み取りは、ソースドレイン間に定電圧をかけた状態で、コントロールゲートに電圧をかけ、流れたドレイン電流を電圧値に変換することによりおこなう。ここで、ドレインソース間に電流が流れるためのコントロールゲート電圧のスレッシュホールド値 $V_{th}$ は図20に示すように、浮遊ゲート中の電荷の有無により変化する。そこで、フラッシュメモリ素子では、この $V_{th}$ の違いをドレイン電流から判断することにより1タ値を読み出しているのである。ここでは、リテンションエラーとは、経年変化によって浮遊ゲートから電荷が抜け落ちることにより発生するエラーをいい、これにより、一定の時間が経過すると、メモリ素子のデータ読み取りエラー率が急激に増加する。

【0006】そこで、従来、特開平3-5995号公報などに記載されているように、フラッシュメモリを用いたファイルシステムにおいては、メモリからのデータ読み取

## 4

りを高信頼化させるため、フラッシュメモリチップ内に符号/復号回路を搭載し、データエラー検出、及びエラー訂正のための冗長データを付加した上で、データを誤り訂正符号(Error Correct Code以下ECC)に変換して記録し、メモリ素子からデータを読み出すときには読み出したECCを用いてデータエラーを検出/訂正する手法が用いられている。

【0007】ここで、このようなシステムで用いられるECCは主として組織符号が用いられている。これは、情報データ部と、冗長データ部が分離されて構成されているECCである。組織符号によれば、情報データ部に冗長データ部を付加することにより、ECCを構成することが出来るため、符号語の中に情報部をそのままの形で取り込むことが出来る。組織符号であるECCには、誤り訂正能力や誤り訂正単位の違いから、ハミング符号、BCH符号、リードソロモン符号等がある。比較的大きい(数百バイト以上)のデータを一括して処理するファイルシステムでは、リードソロモン符号が多く用いられている。これは、このようなファイルシステムではデータの取り扱いの最小単位がバイト(8ビット)となることが多く、データ誤りもバーストエラーが多いため、誤り訂正の最小単位が1ビットであるBCH符号に対して、誤り訂正の最小単位がシンボル(複数ビット)であるリードソロモン符号のほうが、符号効率が良くなるからである。

【0008】一方、一回書き込み可能コンパクトディスク(CD-R)や、デジタルオーディオテープ(R-DAT)など、データ誤り率が比較的大きい(10の-3乗程度)記録再生媒体においては、従来より、ランダムエラー、バーストエラーに対して、強力な誤り訂正能力を持つ積符号が用いられている。これは、情報データをマトリクス単位で扱い、その行方向、列方向に対し、それぞれを符号化するECCである。積符号の特徴として、行方向、列方向のECCが協力して復号を行う事が挙げられる。積符号の復号方法はいろいろ知られており、それぞれ最大誤り訂正能力、復号に要する計算量等に違いがある。

【0009】CD-R、R-DAT等の先に挙げた記録再生媒体では、特開昭63-298776号公報に記載のような、リードソロモン符号を積符号化した方法が主に用いられている。

【0010】

【発明が解決しようとする課題】従来のフラッシュメモリではメモリ1素子に対し、1ビットを対応させていた。このため読み出しの際にドレイン電流から区別すべき $V_{th}$ は1素子あたり2つであり、 $V_{th}$ 間の間隔を充分にとることが出来た。しかし近年、フラッシュメモリを用いたファイルシステムに対する大容量化、低コスト化の要求から、1素子に対して2ビット以上を対応させる必要性が出てきた。このことは読み出しの際にドレイン電流から区別すべき $V_{th}$ が1素子あたり4つ以上になることを示す。このため各 $V_{th}$ 間の間隔は狭くなり、メモリ素子が

らのデータ読み出しエラーは必然的に増加する。

【0011】この場合に、必要とされるビット誤り訂正不能率をメモリチップ単体で満たすためにメモリチップ内部の符号／復号回路に、より強力なECCを用いる事が要求される。

【0012】しかし、このようにすると、符号／復号回路がより複雑になり、復号時間が長くなる。またメモリチップの符号／復号回路規模が大きくなることにより、メモリチップにおける符号／復号回路の占める割合が増加し、メモリ実装面積が減少することにより、多値記録化によるメモリ容量増加のメリットが生かせない。

【0013】また、ファイルシステムとして使用するとき要求されるビット誤り訂正不能率は、メモリチップ単体に必要とされるビット誤り訂正不能率よりも一般に低い。

【0014】そこで、従来のメモリチップ内の、符号／復号回路の外側に更に符号／復号回路を設け、ファイルシステムとしての誤り訂正不能率を低くすることも考えられる。

【0015】しかし、メモリチップ内の符号／復号回路の他に、これと独立して符号／復号を行う符号／復号回路を外側に設けると、ECCに必要な冗長データが増加し、メモリチップにおける情報データの格納効率はきわめて低くなる。

【0016】そこで、本発明は、メモリチップ内の符号／復号回路を大規模化することなく、ファイルシステムなどとして使用される記録再生装置に用いる場合にも、記録再生装置として要求される誤り訂正不能率を満足することのできるメモリチップ及び記録再生装置を提供することを課題とする。

【0017】

【課題を解決するための手段】前記目的達成のために、本発明は、メモリを内蔵したメモリチップを用いた記録再生装置であって、前記記憶再生装置は、前記メモリチップの外部に、記憶対象の情報データを誤り訂正符号化し第1の誤り訂正符号を生成する外符号／復号回路を備え、前記メモリチップは、外符号／復号回路が生成した第1の誤り訂正符号を、さらに、誤り訂正符号化し第2の誤り訂正符号を生成し、前記メモリに記憶する内符号／復号回路を内蔵し、前記内符号／復号回路は、前記メモリより読み出した第2の誤り訂正符号を用いて誤り訂正を行い、前記第1の誤り訂正符号を復号し、前記外符号／復号回路は、前記内符号／復号回路が復号した第1の誤り訂正符号を用いて誤り訂正を行い、前記情報データ復号することを特徴とする記憶再生装置を提供する。

【0018】本記憶再生装置によれば、メモリチップ内部の内符号／復号回路と、メモリチップ外部の内符号／復号回路が各々誤り訂正符号、復号を行う。すなわち、2重に誤り訂正処理が施されるので、メモリチップから読み出したデータの強力な誤り訂正能力を発揮できる。

また、この際、メモリチップ内部の内符号／復号回路は、単独で、本記憶再生装置の誤り不能率を達成する場合に比べ小規模化できる。

【0019】また、このようなメモリチップ内外の誤り訂正の分担によれば、メモリチップを外符号／復号回路を備えていないシステムにおいて使用する場合でも、メモリチップ内部の内符号／復号回路による誤り訂正処理によって従来のデータ誤り率と同程度のデータ誤り率を確保するようメモリチップを構成することが可能となる。

【0020】また、前記外符号／復号回路は、記憶対象の情報データをマトリクス化し、当該マトリクスの各列／行を誤り訂正符号化して、複数列／行の第1の誤り訂正符号を生成し、前記内符号／復号回路は、外符号／復号回路が生成した複数列／行の第1の誤り訂正符号を、行／列方向に並べたマトリクスの、各行／列を誤り訂正符号化して、複数列／行の第1の誤り訂正符号を生成し、当該複数列／行の第1の誤り訂正符号は、前記情報データの積符号を形成するようにすれば、メモリチップ内の符号／復号回路の他に、これと独立して符号／復号を行う符号／復号回路を外側に設ける場合に比べ、同程度の誤り不能率を達成するのに必要な冗長データが少なくてすむ。よって、メモリチップにおける情報データの格納効率を向上することができる。

【0021】

【発明の実施の形態】以下、本発明に係る記録再生装置の一実施形態を、記録媒体としてフラッシュメモリを用い、ファイルシステムとして使用される記録再生装置への適用を例にとり説明する。

【0022】図1に、本実施形態に係るファイルシステムの構成を示す。

【0023】図1においてメモリ103はデータの記録あるいは再生を行う記録媒体、メモリチップ102は、メモリ103と内符号／復号回路104とを含むチップである。外符号／復号回路105はインターフェースLSI106内にあって、外符号生成や外符号検出及びデータの訂正を行う回路である。インターフェースLSI106はフラッシュメモリ103を使用したファイルシステム101におけるシステムバス109とのインターフェース制御を行うLSIである。

【0024】より具体的には、本実施形態に係るファイルシステムは、たとえば、図2に示すように構成することができる。

【0025】図2において、フラッシュメモリ502は、図1のメモリチップ102に相当し、図1のメモリ103に相当する記録媒体と、図1の内符号／復号回路104に相当するECC回路を備えている。インターフェースLSI503はフラッシュメモリ502を使用したファイルシステム501におけるシステムバス506とのインターフェース制御を行うLSIであり、図1のインターフェースLSI106に相当する。インターフェースLSI503内のECC回路5031は図1

の外符号／復号回路105に相当する。

【0026】マイコン504はシステムバスを通じて送られてきた命令を解釈し、解釈結果に応じてフラッシュメモリ502へのデータ読み込み及び書込み(Read/Write、以下R/W)、及びDRAM505へのデータR/Wを制御する中央処理装置(CPU5041)をやROM5042やRAM5043を備えており、ファイルシステム501のコントローラの役割を担っている。DRAM505はフラッシュメモリ502のデータを、インターフェースLSI503内のECC回路5031に渡す際、データのバッファの役目を担う補助メモリである。

【0027】これらの各部は、制御信号線、アドレスバス、データバスによって接続されている。

【0028】ここで、図2のインターフェースLSI503は、たとえば、図3に示すように構成される。

【0029】図3において、システムインターフェース部601はシステムバスを通じて送受する命令及びデータの制御を行う。マイコンインターフェース部602はマイコン504との間で送受する命令及びデータの制御を行う。DRAM制御部603は、DRAM505との間で送受するデータの制御を行う。フラッシュメモリ制御部604は、フラッシュメモリ502へ送られる命令、及びフラッシュメモリ502との間で送受するデータの制御を行う。ECC制御部605はマイコン504の命令により、システムバス506を通して入力するデータや、フラッシュメモリ502からのデータをECC回路5031に渡し、ECC生成や、ECC検出及びデータの訂正といったECC訂正手段の制御を行う。

【0030】以下、このようなファイルシステムにおいて行われるデータ誤り検出、訂正処理について説明する。

【0031】まず、具体的な動作について説明する前に、本実施形態で用いる積符号について説明する。

【0032】本実施形態は、外符号／復号回路105と内符号／復号回路104が協調して、積符号を行う。情報データの記憶の際には、図4に示すフォーマットに従い、外符号／復号回路105は情報データを符号化して外符号C1を生成し、内符号／復号回路104は、外符号を符号化して、内符号C2を生成し、メモリ103に記憶する。

【0033】本実施形態では、情報データのR/Wは1記録再生データ領域512バイト単位で行う。また、積符号化においては、複数の記録再生データ領域を一括して符号化してもよいが、本実施形態では1記録再生データ

領域ごとに積符号化を行う。積符号に用いられるECCは、内符号C2、外符号C1ともリードソロモン符号を用いる。内符号C2の訂正能力は1シンボルとする。外符号C1の訂正能力は2シンボルとし、内符号／復号回路104からのイレージャフラグ信号をもとにイレージャ訂正を行う。イレージャ訂正とはイレージャフラグ信号を利用することで、外符号C1のみを用いて誤り訂正を行う場合よりも誤り訂正能力を向上させる周知の訂正法である。なお、イレージャ訂正については、たとえば、日刊工業新聞社発行の「デジタルビデオ記録技術」の89ページから122ページや、共立出版発行の符号理論などに記載されている。

【0034】さて、外符号C1のみを用いた場合の誤り訂正能力を $t$ 2シンボルとすると、イレージャ訂正の訂正能力は最大で $2 \times t$ 2シンボルとなる。本実施形態では $t$ 2は2なので、訂正能力は最大で4シンボルとなる。

【0035】このような積符号の復号方式は、いろいろな方式があるが、本実施形態では先に内符号／復号回路104で内符号C2を復号した後、外符号／復号回路105で外符号C1を復号する。積符号においては、情報データが正方マトリクスデータに近いほど、符号化効率が良いため、情報データは16バイト×32バイトのマトリクスデータとして扱う。外符号C1は図4に示す32バイトの情報データの列データを符号化したものであり、内符号C2は16バイトの情報データもしくは外符号／復号回路105で付加された冗長シンボルの行データを符号化したものになる。

【0036】内符号で符号化する情報量は1行あたり $16 \times 8 = 128$ ビット、外符号で符号化する情報量は1列あたり $32 \times 8 = 256$ ビットになる。またリードソロモン符号のシンボル長は、1バイトが1シンボルに対応していた方がデータの取り扱い上便利であるので、本実施形態では、1シンボルのビット数8とし、冗長シンボル長は内符号16シンボル、外符号32シンボルのリードソロモン符号を用いる。

【0037】ここで、リードソロモン符号における最大符号語長、及び最大情報量についての条件を式1に示す。

【0038】

【数1】

誤り訂正能力を $t[\text{symbol}]$ とすると

符号語がガロア体 $GF(2^m)$ 上の元であるとき

最大符号語長 $=2^m - 1$  [symbol]

冗長シンボル数 $=2t$  [symbol]

最大情報記号数 $=2^m - (2t + 1)$  [symbol]

$1[\text{symbol}] = m$  [bit]

符号語 $W: (w_1, w_1, \dots, w_{n-1})$ とすると

符号語多項式 $C(x): w_{n-1}x^{n-1} + w_{n-2}x^{n-2} + \dots + w_1x + w_0$

$C(x)$ は $GF(2^m)$ 上の元 $1, \alpha, \alpha^2, \dots, \alpha^{2^t-1}$ を解に持つ。

#### 式1: リードソロモン符号における 最大符号語長、最大情報記号数の条件

【0039】 $t$ はシンボル訂正能力である。ガロア体は、リードソロモン符号の符号語が含まれる集合で、 $m$ は1シンボルあたりのビット数に対応する。 $m$ が大きいほど、最大符号語長及び最大情報量が大きくなる。しかし、1シンボルあたりに含まれるビット数が多くなるため、ビットあたりの誤り率が同じなら、訂正能力 $t$ が同じでも、冗長ビット数が増加する。ここで、リードソロモン符号の符号語を $W$ とすると、符号多項式 $C(x)$ は、符号語 $W$ の各成分を係数に持つ多項式として定義される。

【0040】本実施形態の場合、シンボル長の条件を満たす $m$ の最小値は8である。この時最大符号語長は、2の8乗 $-1 = 255$ シンボル、最大情報量は1シンボル訂正時で2の8乗 $-3$ バイト $= 253$ バイト $= 2024$ ビット、2シンボル訂正時で2の8乗 $-5 = 251$ バイト $= 2008$ ビットとなり、条件を満たす。従ってリードソロモン符号の符号語の集合は内符号、外符号とも2の8乗のガロア体を用いる。冗長シンボル長は内符号で、2シンボル $= 2$ バイト、外符号で4シンボル $= 4$ バイトとなる。

【0041】以下、このような積符号を生成しメモリ105に記録する具体的動作について説明する。

【0042】図5に、メモリ105への記録処理の手順を示す。

【0043】情報データは、システムバスを109を介して記録再生データ領域、つまり512バイトごとに、図4の行順に入力される。

【0044】インターフェースLSI106内の外符号／復号回路105は情報データを16バイト $\times$ 32バイトのマトリクスデータに変換する。しかし、実際にメモリ103に記録される時は、情報データは512バイト

20 の列データとして記録されるため、マトリクスデータへの変換は必ずしも必要ではなく、1次元配列データのままで扱ってもよい。この場合、メモリへの記録処理はステップ701からはじめてもよい。以下の説明では、情報データは1次元配列データとして扱い、ステップ701から処理を行う場合を例にとる。

【0045】外符号／復号回路105は、マトリクスデータあるいは1次元配列データをステップ701に示すように情報データを外符号C1に符号化する。更に、ステップ702に示すように、外符号化された1次元配列データの構成を変換し内符号回路104に入力する。ステップ701において512バイトの1次元配列データを外符号C1に変換する処理、及びステップ702において、外符号化された1次元配列データの構成を変換する処理の様子を図6に示す。

【0046】最初に、512バイトの1次元配列データを外符号C1に変換する処理では、インターフェースLSI106内の外符号／復号回路105は512バイトの情報データを一旦、内部のメモリに順番に格納した後、内部のメモリの最初のアドレスからデータアドレスを15バイトずつスキップしながら情報データを1バイトずつ読み取る。これを32回繰り返し、32バイトの列データを生成する(a)。この操作を、最初に情報データを読み取るアドレスを1増加させながら、16回すなわち16バイト行データの各列について行い、列データ16個を生成する。

【0047】この列データを各々リードソロモン符号化して、情報データ部を外符号C1に符号化する。外符号／復号回路105の訂正能力は2シンボルだから必要な冗長シンボル長は4シンボルとなる。冗長バイト数は、 $4 \times 8 \div 8 = 4$ バイトになる。このため外符号C1の冗

長データ部 R1 は、 $4 \times 16 = 64$  バイトの情報量となる。

【0048】次に外符号化された 1 次元配列データの構成を変換する処理について説明する。内符号／復号回路 104 はデータを図 4 の行方向に符号化するため、あらかじめ外符号／復号回路 105 において、4 バイト（1 行 4 列） $\times$  16 個からなる外符号の冗長部 R1 を 16 バイト（16 行 1 列） $\times$  4 行のデータに変換しておく。

【0049】この変換は、各列について生成した外符号 C1 の冗長部 R1 を順番に内部のメモリに格納した後、この内部のメモリの、冗長データ部 R1 を格納した最初のアドレスから冗長データ部 R1 を、データアドレスを 3 バイトずつスキップしながら 1 バイトずつ 16 回繰り返して読み出し、16 バイトの行データを生成する

(b)。この操作を最初の冗長バイトを読み出すアドレスを 1 増加させながら、4 回繰り返して得られた 4 つの行データを、図 6 の情報データ部の後ろに順番に付加する。このようにして生成した 32 行の情報データ部と、その後ろの 4 行の冗長データ部 R1 の 1 次元配列を、内符号／復号回路 104 に外符号 C1 として出力する。

【0050】図 5 に戻り、次のステップ 703 では、メモリチップ 102 内の内符号／復号回路 104 が、入力された外符号 C1 を内符号 C2 に変換する。更に、ステップ 704 に示すように、内符号化された 1 次元配列データの構成を変換しメモリ 103 に記録する。

【0051】ステップ 703 において外符号 C1 を内符号 C2 に変換する処理、及びステップ 704 において、内符号化された 1 次元配列データの構成を変換する処理の様子を図 7 に示す。

【0052】最初に外符号 C1 を内符号 C2 に変換する処理では、内符号／復号回路 104 は図 7 に示すように、512 バイトの情報データ部 + 64 バイトの外符号 C1 の冗長データ部 R1、つまり  $(32 + 4)$  個  $\times$  16 バイトの 1 次元配列データを内符号／復号回路 104 内部のメモリに一旦記録した後、内符号 C2 に変換する。

【0053】1 次元配列データは情報データ部、冗長データ部 R1 の順に図 4 の行順に入力される (a)。内符号／復号回路 104 は、最初に情報データ部を 16 バイトごとにリードソロモン符号化し、内符号 C2 に符号化する。次に外符号 C1 の冗長データ部 R1 を 16 バイトごとにリードソロモン符号化し、内符号 C2 に符号化する。

【0054】ここで、内符号／復号回路 104 の訂正能力は 1 シンボルだから、必要な冗長シンボル長は 2 シンボルである。冗長バイト数は  $2 \times 8 \div 8 = 2$  バイトになる。従って内符号 C2 の冗長データ部 R2 は、 $2 \times (32 + 4) = 72$  バイトの情報量となる。

【0055】こうして積符号化された情報データを、図 5 のステップ 704 において、1 次元配列データとしてメモリ 103 に記録する。

【0056】メモリ 103 に記録される積符号のデータ格納フォーマットを図 8 に示す。

【0057】メモリ 103 の 1 行は、512 バイトの情報データ部と、64 バイトの外符号 C1 の冗長データ部 R1 + 72 バイトの内符号 C2 の冗長データ部 R2 + R/W に関するアクセスデータが記録された管理データ部から構成されている。

【0058】フラッシュメモリを使用する場合、その構造上、1 行の部分的な R/W は困難なため、情報データ部と管理データ部とからなる 1 次元配列データは一括して記録される。情報データと管理データ部は、物理的あるいは論理的なパーティションによって分離して管理する。

【0059】管理データ部は、図 7 に示すように、内符号／復号回路 104 が、外符号／復号回路 105 から送られた外符号 C1 (a) の後ろに (b) に示すように、各行について生成した内符号 C2 の冗長部 R2 を付加し、更にアクセスデータが付加された構成となっている。ただし、この順序は任意としてもよい。

【0060】次に、このようにしてメモリ 105 に記録された積符号を復号する処理について説明する。

【0061】図 9 にこの処理の処理手順を示す。

【0062】ステップ 801 に示すようにメモリ 103 から読み出された積符号は、図 10 (a) に示すように、内符号／復号回路 104 に、外符号 C1 (512 バイトの情報データ部 + 64 バイトの外符号 C1 の冗長データ部 R1) + 72 バイトの内符号 C2 の冗長データ部 R2 からなる 1 次元配列データとして入力される。

【0063】内符号回路／復号回路 104 は、ステップ 802 に示すようにこの 1 次元配列データを順番に一旦内部のメモリに記憶した後、18 バイトの内符号 C2 を 36 行生成する。これは内部のメモリから 2 バイトを読み、情報データまたは冗長データ部 R1 に付加する (図 10 (b)) 事で生成する処理を、36 回繰り返す事により行う。

【0064】次に内符号 C2 を生成したら、ステップ 803 からステップ 806 において内符号 C2 を用いて復号処理をおこなう。図 8 では点線で囲まれたステップ群が 2 つあるが、上の方が内符号 C2 の復号処理に対応する。

【0065】内符号 C2 の復号処理においては、最初にステップ 803 に示すように、各行、すなわち 18 バイトの内符号 C2 36 個についてシンドローム S(x) を計算し、これから図 4 の行データのエラー訂正及び検出を行う。

【0066】シンドローム S(x) とは符号語に発生した誤りの状態を示すパターンである。このパターンは記録符号語に関係なく符号語内に発生した誤りのみで決定される。シンドロームの定義を式 2 に示す。読取り符号語を R(x) とすると、誤り系列 E(x) = 0 の時は読



み込んだデータに誤りが無い事を示す。このときR  
 $(x) = C(x)$  となるから、式2の定義から、S  
 $(x) = 0$  となる。誤り系列E(x)が非零の場合、読  
 み込んだデータに誤りが発生した事を示す。このときS  
 数2

(x)は非零となり、式2で定義するような連立方程式  
 になる。

【0067】

【数2】

符号語  $= C(x)$

読み取り語  $R(x) = C(x) + E(x)$

ただし、E(x)：誤り系列

式1の条件からC(x)は、 $1, \alpha, \alpha^2, \dots, \alpha^{2t-1}$ を解に持つ。

この時シンδροーム  $S(\alpha^i)$  ( $i = 0, 1, 2, \dots, 2t-1$ )

$= E(\alpha^i)$

$E(x) = 0$  なら  $S(x) = 0$

## 式2：シンδροームの定義

【0068】リードソロモン符号の場合、誤り訂正単位  
 はシンボル（複数ビット）なので、S(x)は誤りシン  
 ボルの位置と誤りの大きさ（シンボル内のビット誤り位  
 置に対応する）の情報を含んでいる。このときS(x)

数3

は式3で定義されるように誤り位置と誤りの大きさで表  
 される。

【0069】

【数3】

リードソロモン符号の場合誤り位置  $\alpha$  の他に

誤りの大きさ  $E_p$  を求めることになるので

$E(x) \neq 0$  なら  $S(x)$  は  $\alpha$  と  $E_p$  に関する連立方程式になる。

符号語を  $C(x)$  ( $1, \alpha, \alpha^2, \dots, \alpha^{2t-1}$ を解に持つ。)

誤り位置を  $(i, j, \dots)$ 、誤りの大きさを  $(E_{p_i}, E_{p_j}, \dots)$

とするとシンδροーム  $S(x)$  は

$$\begin{cases} S(x) = S_i(\alpha^i, E_{p_i}) \\ S_0 = E_{p_i} + E_{p_j} + \dots \\ S_1 = \alpha^i E_{p_i} + \alpha^j E_{p_j} + \dots \\ S_2 = \alpha^{2i} E_{p_i} + \alpha^{2j} E_{p_j} + \dots \\ \vdots \\ S_{2t-1} = \alpha^{(2t-1)i} E_{p_i} + \alpha^{(2t-1)j} E_{p_j} + \dots \end{cases}$$

## 式3：リードソロモン符号のシンδροームの定義

【0070】Nデータ内のエラー数が訂正能力t以内な  
 ら、式3に示すS(x)から誤り位置と誤りの大きさを見  
 つける事で、符号誤りを訂正することができる。エラ  
 ー数が訂正能力を超える場合は、式3の連立方程式の解

が範囲外となるか不定となり、誤り訂正はできない。こ  
 の場合、誤りパターンによっては誤り検出が可能な時が  
 ある。しかし誤り方によっては、誤った符号語を推定し  
 てしまう（誤訂正）ことがある。本実施形態の内符号C

2の場合、誤り訂正能力 $t$ は1である。

【0071】シンドローム $S(x)$ の計算が終了したならば、次にステップ804に示すように $S(x)$ の値によって誤り訂正／検出を行うかどうかを判定する。

【0072】そして $S(x) = 0$ の場合は、その内符号C2の行に誤りがない事を示すので、内符号／復号回路104は、その内符号の行から、内符号C2の2バイトの冗長データ部R2を除去し16バイトの内符号C2を訂正せずに外符号／復号回路105に出力する。

【0073】一方、 $S(x)$ が非零の場合は、その内符号C2に誤りが発生した事を示すので、次にステップ805において、符号語が訂正可能かどうかをシンドロームを用いて計算する。そして、内符号C2から計算されたシンドロームパターンがある特定の符号語のシンドロームパターン群に一致した場合はステップ806で誤り訂正を行い、その内符号C2の行から内符号C2の2バイトの冗長データ部R2を除去し、訂正した16バイトの内符号C2を外符号／復号回路105に出力する。一致しない場合は、内符号C2の訂正能力を超えるエラーが発生したとみなし、誤り検出処理のみを行い、その内符号の行から、内符号C2の2バイトの冗長データ部R2を除去し16バイトの内符号C2を訂正せずに外符号／復号回路105に出力する。

【0074】このとき、外符号C1による誤り訂正時において内符号C2に誤訂正に起因する外符号C1の誤訂正を少なくするため、内符号C2の復号時において最大訂正能力以上の誤りが発生した場合、つまり本実施例の場合 $S(x)$ が非零の場合は、内符号／復号回路104はステップ807において、誤り訂正／検出もしくは誤り検出のみを行った行の全てのシンボルにイレージャフラグ情報を付加し、外符号／復号回路105に出力する。

【0075】いま、図11に示す積符号の構成図の右横の網掛け三角形で示されている行が、内符号C2の訂正能力を超える誤りが発生した行を示すものとする。1、2、6行目はランダムエラーが発生していることを示す。4行目はバーストエラーが発生していることを示す。この場合、1、2、4、6行目にあたる16バイトのデータ全てにイレージャフラグ情報を付加する。イレージャフラグ情報は内符号復号データとは別に、付加の対象となったバイトを識別可能なように外符号／復号回路105に出力する。

【0076】次に、ステップ808では、外符号／復号回路105において内符号C2によって復号された外符号C1を、36バイトごとの列に構成し、ステップ809からステップ814において外符号C1を用いて復号処理をおこなう。図8では点線で囲まれたステップ群が2つあるが、下の方が外符号C1の復号処理に対応する。

【0077】最初にステップ808について説明する。

【0078】外符号／復号回路105には、外符号C1が図12aに示すように、情報データ部の後に冗長データ部R1が付加された形態の1次元配列データとして入力される。

【0079】外符号／復号回路105は、この1次元配列データを一旦内部のメモリに格納した後、データアドレスを15バイトずつスキップしながら1バイト読み取る処理を36回繰り返して36バイトの外符号C1の列データを生成する処理を、読み出しを開始する先頭のアドレスを順次1バイトずつ進めながら16回繰り返して、16列の外符号C1を生成する。

【0080】次に外符号／復号回路105は、外符号C1の各列について、ステップ810に置いてシンドローム $S(x)$ を求める。そして $S(x) = 0$ の場合は、読み出された1次元配列データに誤りがないことを示すので、外符号／復号回路105は、その外符号の列から、冗長データ部R1を除去し、32バイトの情報データを訂正せず出力する。

【0081】一方 $S(x) \neq 0$ でない場合は、次に、情報データ部+外符号C1の冗長データ部R1からなる1列36バイト中のイレージャフラグが付加されたバイトの総数を $E_r(x)$ として、ステップ811において $E_r(x)$ の値を判定する。

【0082】そして、 $E_r(x) = 0$ の場合は、ステップ813においてイレージャフラグを用いた訂正を行わず、外符号C1のみによる通常の復号を行う。

【0083】一方 $E_r(x)$ が非零の場合は、ステップ812において、 $E_r(x)$ の数に応じて、誤り訂正／検出を行う。すなわち、 $E_r(x) > 4$ の場合、外符号C1の訂正能力を超える誤りが生じたことを示すので、ステップ814に示すように誤り訂正不能とする。ただし、イレージャフラグから計算されたシンドローム $S(x)$ の連立方程式が全て0である場合は、例外的に誤りなしとして、その外符号の列から冗長データ部R1を除去し、32バイトの情報データを訂正せず出力する。

【0084】 $E_r(x) \leq 4$ である場合、誤り訂正が可能となる。

【0085】 $E_r(x) \leq 4$ である場合のイレージャ訂正における誤り訂正は次のように行う。例えば図10の7列目、すなわち下部において白抜き三角形で示されている列は、イレージャフラグが列データ内に2つ以下である。この場合はイレージャ訂正を用いても誤り訂正能力は2しかない。従って通常どおり外符号C2を用いてリードソロモン復号による誤り訂正を行い、その外符号の列から冗長データ部R1を除去し、訂正した32バイトの情報データを出力する。

【0086】次に、例えば、8列目、すなわち下部において灰三角形で示されている列は、イレージャフラグが列データ内に3つある。 $t=2$ が2の場合、得られるシンドローム $S(x)$ の連立方程式は式2の条件から4つで

ある。従ってシンドロームから誤りの大きさに関する式が3つ、イレージャフラグから得られた誤り位置以外の誤りの有無についての判別式が1つ得られる。そこで、イレージャフラグから得られた誤り位置以外に誤りがない場合は誤り訂正を行い、その外符号の列から冗長データ部R1を除去し、訂正した32バイトの情報データを出力し、イレージャフラグから得られた誤り位置以外に誤りがある場合は、誤り検出のみを行い、その外符号の列から冗長データ部R1を除去し、32バイトの情報データを訂正せず出力する。

【0087】ついで6列目、すなわち黒三角形で示されている列はイレージャフラグが列データ内に4つある場合を示す。シンドロームからは誤りの大きさに関する式が4つ得られる。外符号／復号回路105では、その外符号の列から冗長データ部R1を除去し、訂正した32バイトの情報データを出力する。

【0088】ただしこの場合、訂正能力を最大に用いて訂正を行うことになるため、イレージャフラグから得られた誤り位置以外の誤りの有無についての判別式が得られず、イレージャフラグから得られた誤り位置以外に誤りがある場合は、誤訂正をする。そこで、ファイルシステムが必要とする仕様にあわせて、高信頼な誤り訂正を行うために、最大訂正を行わず、 $m - \text{out-of-}n$ イレージャ訂正( $n > m$ )を行うようにしてもよい。これはイレージャフラグが正しいか否かを検定し、 $n$ 個のイレージャフラグが検出されても、その中で真の誤りシンボル数が、 $m$ 個以下のときのみ誤りを訂正するものである。例えば、 $3 - \text{out-of-}4$ イレージャ訂正を行うこととし、6列目のような場合には、誤り検出のみを行い、その外符号の列から冗長データ部R1を除去し、32バイトの情報データを訂正せず出力するようにしてもよい。

【0089】ここで、情報データ部はシステムバス109を介して出力される。

【0090】以上、本実施形態に係るファイルシステムについて説明した。

【0091】以下では、メモリチップ102を外符号／復号回路105と組み合わせずにエラー訂正／検出に関して単体で使用する場合について説明する。

【0092】図13は、メモリチップ102の構成を示したものである。メモリ103はデータの記録あるいは再生を行う記録媒体、メモリチップ102は、メモリ103と内符号／復号回路104を含む。内符号／復号回路104は図1の内符号／復号回路と同じ回路である。

【0093】図14に、この場合のメモリ103への記録処理の手順を示す。

【0094】メモリチップ102単体で使用する場合、システムバス109を通じて入力された情報データは1記録再生データ領域、つまり512バイトごとに1次元配列データとして内符号／復号回路104に入力され

る。

【0095】メモリチップ102内の内符号／復号回路104は、まず、ステップ1601においてマトリクスデータあるいは1次元配列の情報データを内符号C1に変換する。すなわち図15aに示すように内符号／復号回路104は、512バイトの情報データを一旦内部のメモリに記録した後、512バイトの情報データから16バイトの行データを生成し内符号C2に符号化する。1シンボル訂正の場合、必要な冗長シンボル長は2シンボルである。従って冗長バイト数は、1行あたり、 $2 \times 8 \div 8 = 2$ バイトになる。従って1記録再生データ領域に対する内符号C2の冗長データ部R2の情報量は $2 \times 32 = 64$ バイトとなる。

【0096】次に、内符号／復号回路104はステップ1602においてデータ構成を変換する。すなわち、内符号C2を図15bに示すように、各行の情報データ部の後に各行の冗長データ部R2とアクセスデータを付加した1次元配列の形態に変換される。

【0097】こうして符号化された情報データを、図5のステップ1603において、1次元配列データとしてメモリ103に記録する。

【0098】次にメモリ103に記録された内符号C2を復号する処理について説明する。

【0099】この処理の処理手順を図17に示す。

【0100】まず、ステップ1701において、メモリ103から1行分の内符号C2を内符号回路／復号回路104に読み出す。図18に示すように、内符号／復号回路104は、入力された512バイトの情報データ+64バイトの内符号冗長データR2からなる1次元配列データaから、18バイトの内符号C2の行を32個生成する。

【0101】これは、一旦1次元配列データを内部のメモリに記録した後、16バイトの情報データを読むごとに内符号C2の冗長データ部R2の先頭から2バイトを読み、16バイト情報データに付加する(b)処理を、32回繰り返すことにより行う。

【0102】次に内符号C2を生成したら、ステップ1703からステップ1706において内符号C2を用いて復号処理をおこなう。図17では点線で囲まれたステップ群が内符号C2の復号処理に対応する。

【0103】内符号C2の復号処理においては、最初にステップ1703に示すように、生成された内符号C2の各行についてシンドローム $S(x)$ を計算する。次いでステップ1704に示すように $S(x)$ の値によって誤り訂正／検出を行うかどうかを判定する。

【0104】ステップ1704において、 $S(x) = 0$ の場合は、読み出された行に誤りがない事を示すので、内符号／復号回路104は、その行から冗長データR2を除去し16バイトの情報データ部を訂正せず出力する。

【0105】一方、 $S(x)$  が非零の場合は、その行に誤りが発生したことを示すので、この場合はステップ1705に示すようにシンドロームを用いて、符号語が訂正可能かどうかを判定する。そして、内符号C2から計算されたシンドロームパターンが、ある特定の符号語のシンドロームパターン群に一致した場合はステップ1706において、誤り訂正を行い、その行から冗長データ部R2を除去し、訂正した16バイトの情報データ部を出力し、一致しない場合は、内符号C2の訂正能力を超えるエラーが発生したとみなし、誤り検出処理のみをおこない、その行から冗長データ部R2を除去し16バイトの情報データ部を訂正せず出力する。またこの際、先の場合と同様に、イレージャフラグ情報を訂正不能信号として出力する。

【0106】以上、メモリチップ102を外符号/復号回路105と組み合わせず使用する場合について説明した。

【0107】以上の説明より理解されるように、本実施形態によれば、メモリチップを、外符号/復号回路を設けたファイルシステム101の記憶媒体として使用した場合、強力な誤り訂正能力を発揮でき、メモリ103のメモリ素子の多値記憶化に伴うデータ読み出しエラー率の増加に対しても、十分な誤り訂正能力を持たせることが出来る。

【0108】また、メモリチップ102を外符号/復号回路を備えていないシステムにおいて使用する場合でも、従来のデータ誤り率と同程度のデータ誤り率を確保することが可能となる。

【0109】また、本実施形態では、情報データ部分と管理データ部とを分離して記録するので、内符号/復号回路及び外符号/復号回路の採用する符号化方式を用途に応じて用意に変化させることができる。さらに外符号/復号回路のみで強力な誤り訂正符号を符号/復号した場合に比べ、復号時間、メモリ使用効率の面でも優位性がある。

【0110】なお、本実施形態の、メモリチップ内/外で処理される内符号/外符号を用いる方式は、積符号を用いる誤り訂正符号の方式訂正能力を変化させることで、ファイルシステム101及びメモリチップ102に求められる、様々な要求仕様を満たすことが可能な方式である。

【0111】なお、本実施形態に係るファイルシステムは、デジタルカメラや携帯情報端末機器用の記憶装置などにも適用することができる。この場合、メモリチップ102は、デジタルカメラや携帯情報端末機器用に脱着可能な可搬型記憶媒体、たとえば、フラッシュメモリを収容したカード型の記憶媒体であるフラッシュメモリカードなどを構成し、外符号/復号回路は、デジタルカメラや携帯情報端末機器本体側に設けるようにする。

【0112】

【発明の効果】以上のように、本発明によれば、メモリチップ内の符号/復号回路を大規模化することなく、ファイルシステムなどとして使用される記録再生装置に用いる場合にも、記録再生装置として要求される誤り訂正不能率を満足することのできるメモリチップ及び記録再生装置を提供することができる。

【図面の簡単な説明】

【図1】ファイルシステムの構成を示すブロック図である。

10 【図2】ファイルシステムの具体的構成例を示すブロック図である。

【図3】インターフェースLSIの構成を示すブロック図である。

【図4】ファイルシステムにおける積符号の構成を示す図である。

【図5】ファイルシステムにおける符号化処理を示すフローチャートである。

【図6】ファイルシステムにおける外符号への符号化のようすを示した図である。

20 【図7】ファイルシステムにおける内符号生成への符号化のようすを示した図である。

【図8】ファイルシステムにおけるメモリ上の記憶フォーマットを示す図である。

【図9】ファイルシステムにおける復号の処理を示したフローチャートである。

【図10】ファイルシステムにおける復号時の内符号への変換のようすを示した図である。

【図11】ファイルシステムにおけるイレージャフラグとイレージャ訂正対象を示す図である。

30 【図12】ファイルシステムにおける復号時の外符号への変換のようすを示した図である。

【図13】メモリチップの構成を示すブロック図である。

【図14】メモリチップ単体使用時の符号化処理を示すフローチャートである。

【図15】メモリチップ単体使用時の内符号への符号化のようすを示した図である。

【図16】メモリチップ単体使用時のメモリ上の記憶フォーマットを示す図である。

40 【図17】メモリチップ単体使用時の復号処理を示すフローチャートである。

【図18】メモリチップ単体使用時の内符号への変換のようすを示した図である。

【図19】フラッシュメモリ素子の構成を示した図である。

【図20】フラッシュメモリ素子の浮遊ゲート電荷とドレイン電流とコントロール電圧の関係を示した図である。

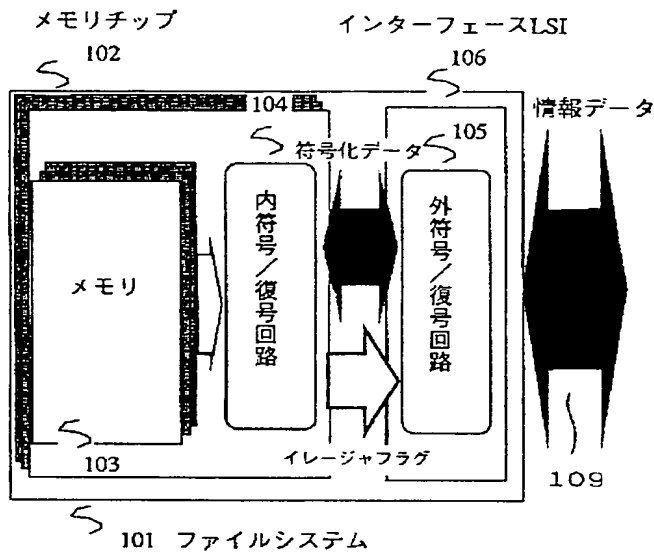
【符号の説明】

50 101 ファイルシステム

21  
 102 メモリチップ  
 103 メモリ  
 104 内符号／復号回路  
 105 外符号／復号回路  
 106 インターフェースLSI  
 109 システムバス

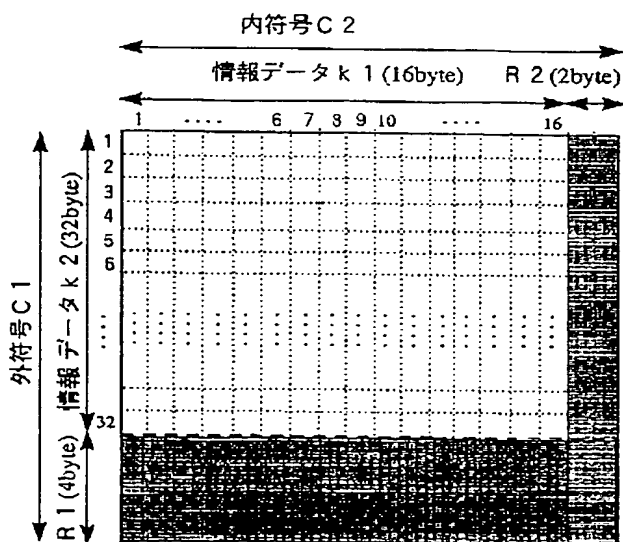
【図1】

図1



【図4】

図4

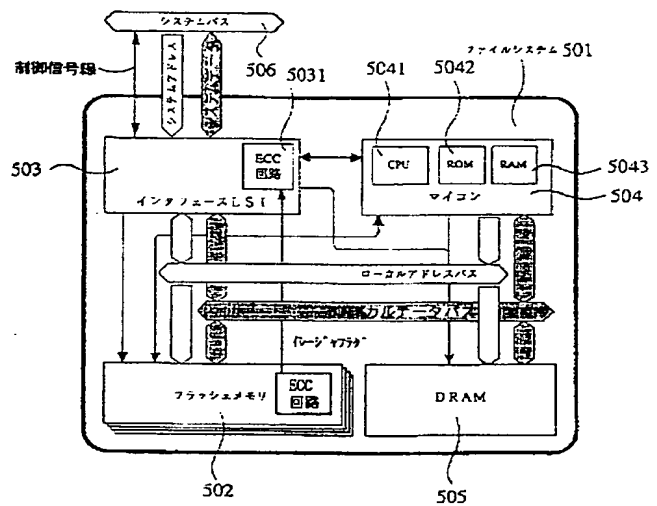


22

501 ファイルシステム  
 502 フラッシュメモリ  
 506 システムバス  
 504 マイコン  
 5031 ECC回路

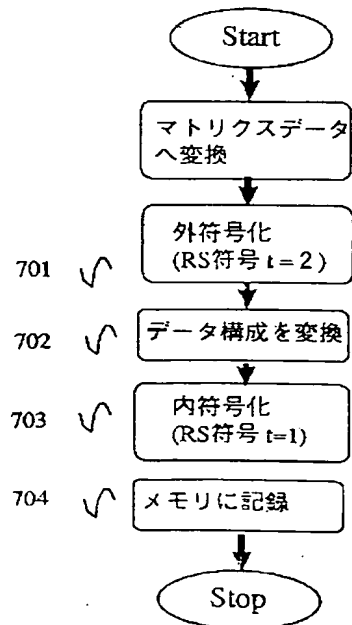
【図2】

図2

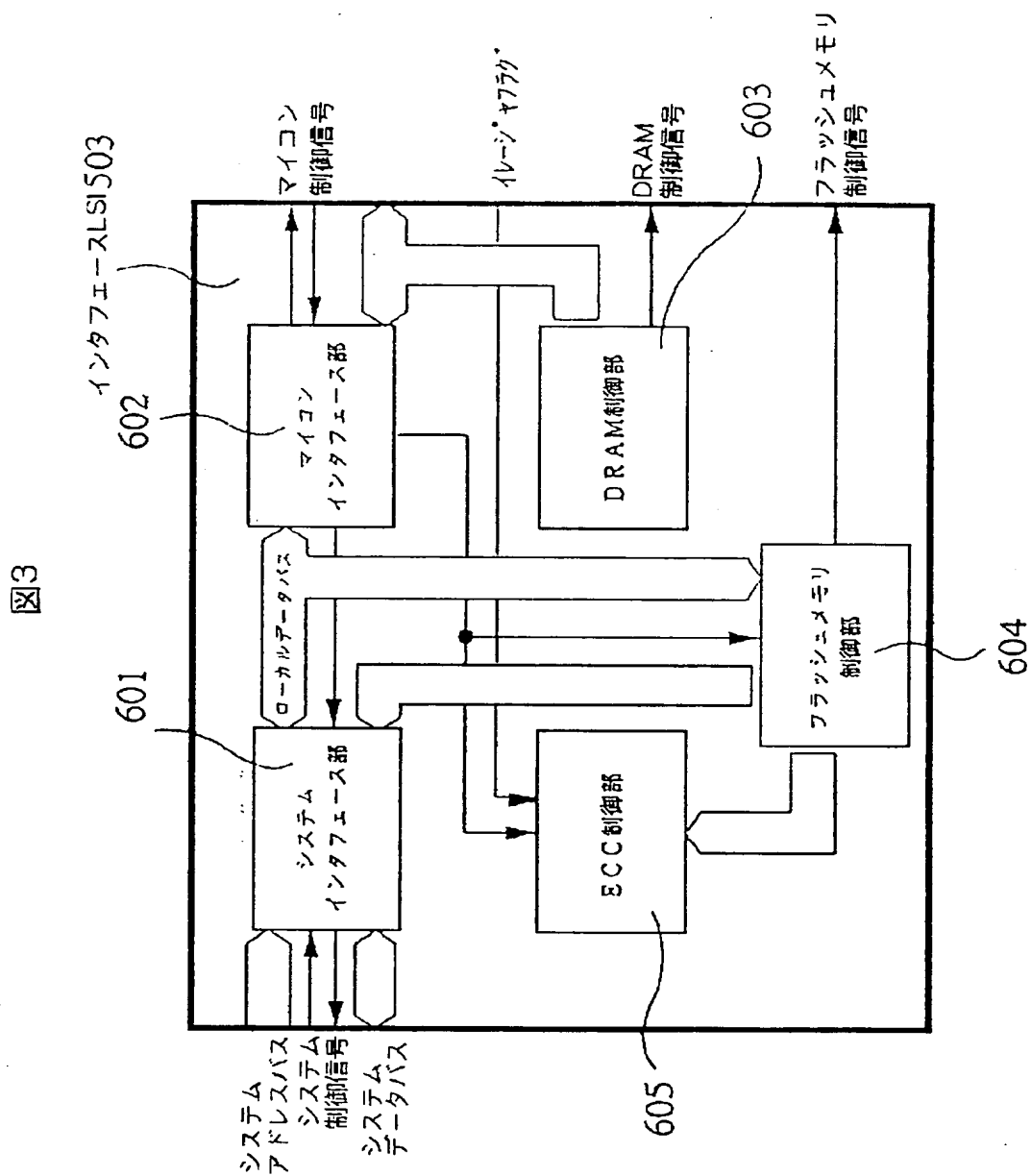


【図5】

図5



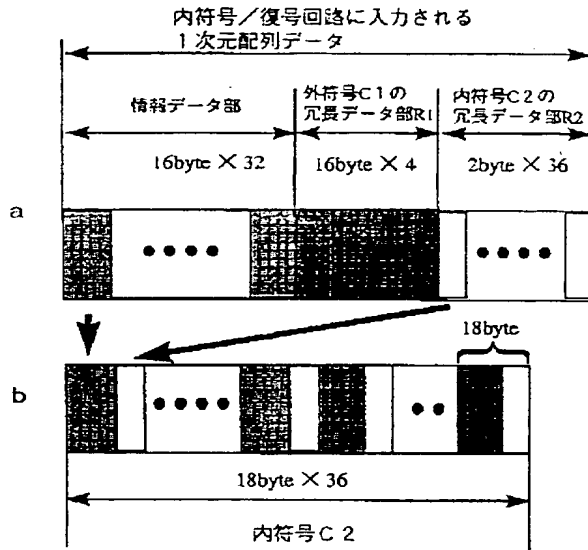
【図3】





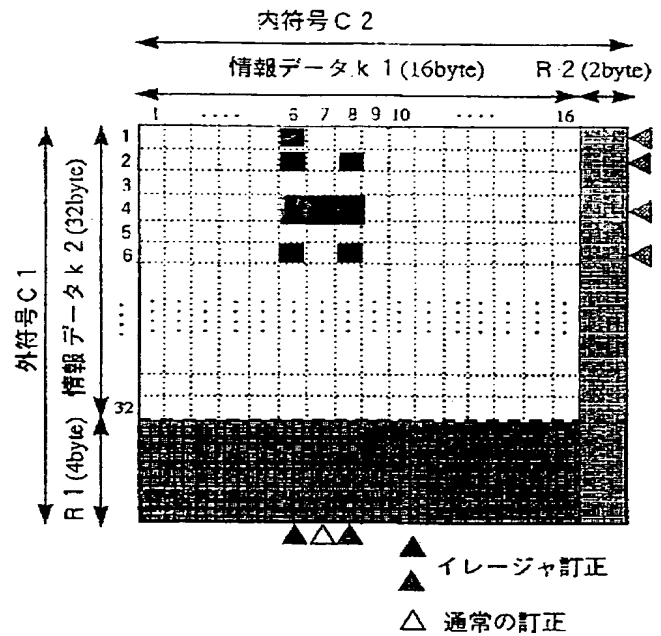
【図10】

図10



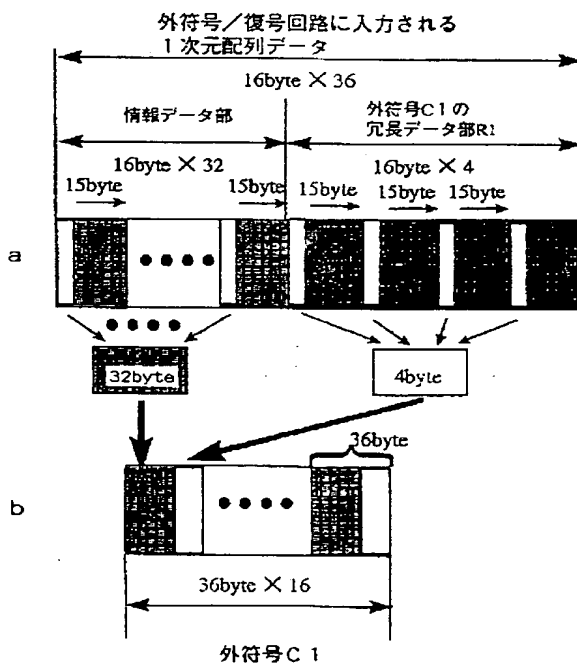
【図11】

図11



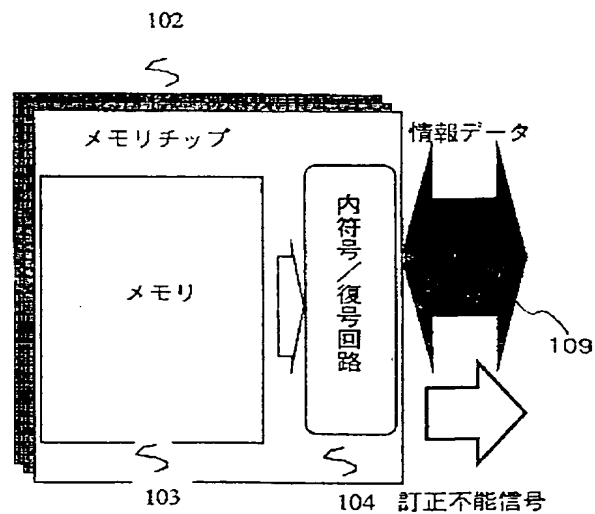
【図12】

図12



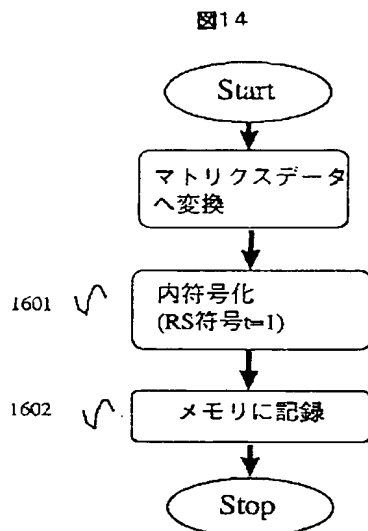
【図13】

図13

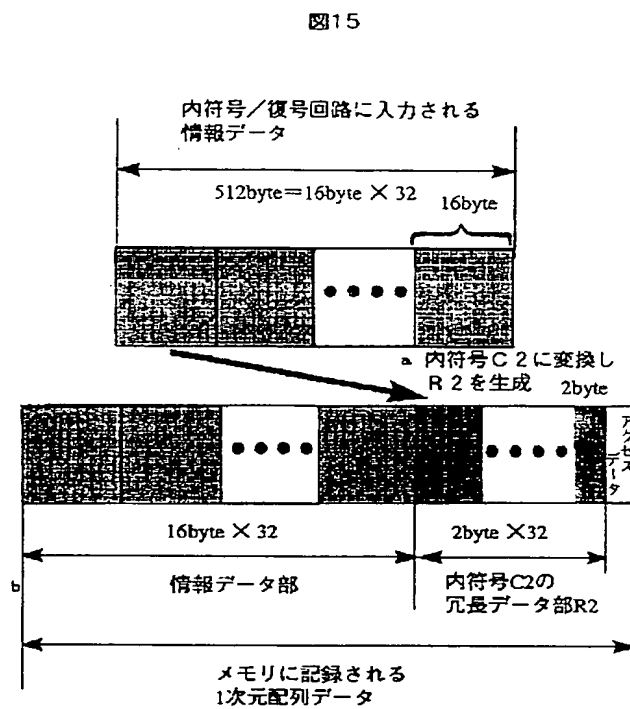




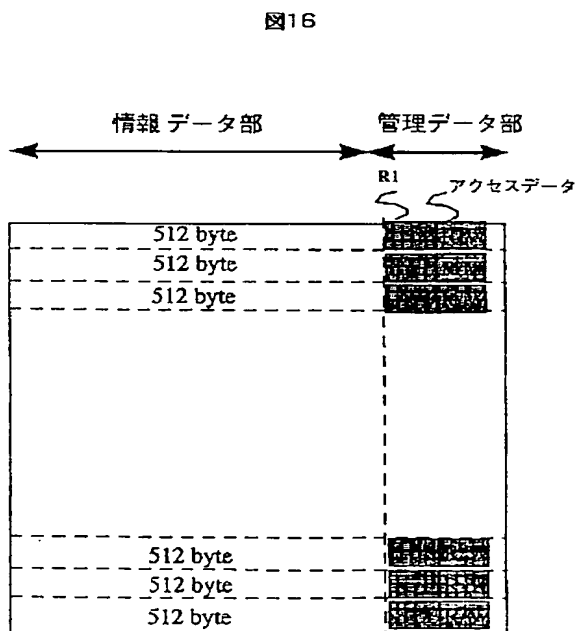
【図 1 4】



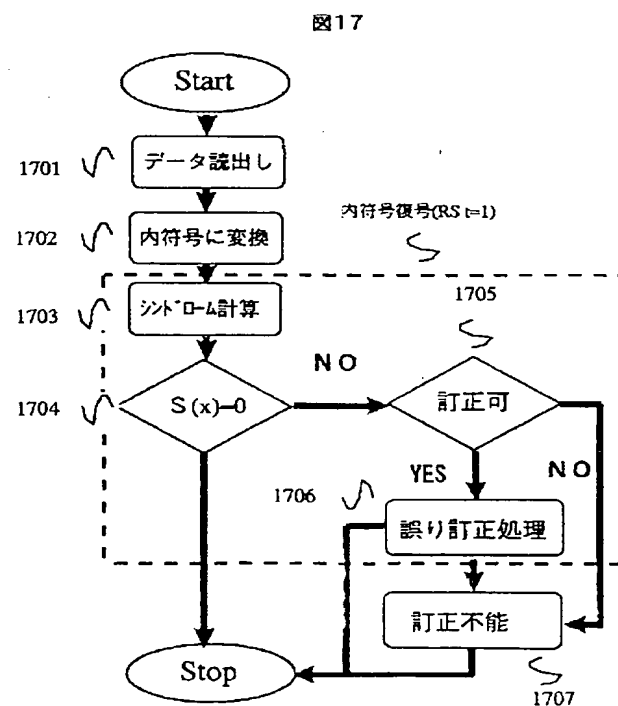
【図 1 5】



【図 1 6】

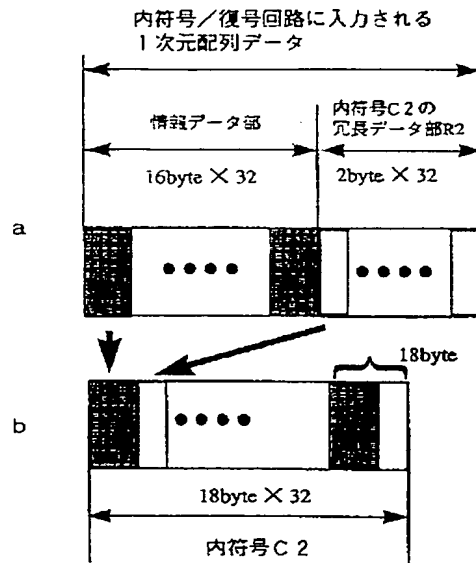


【図 1 7】



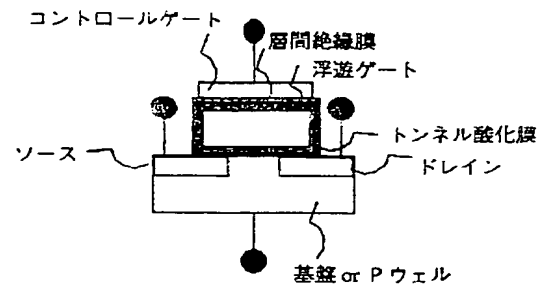
【図 18】

図18



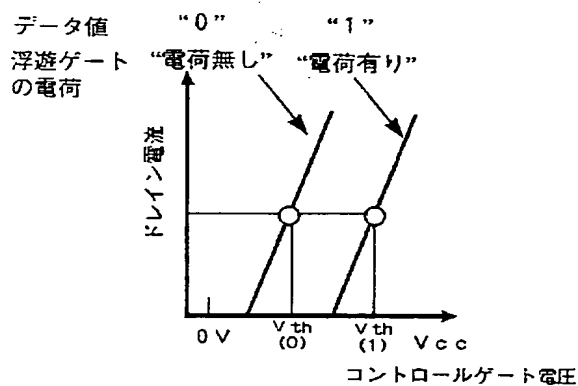
【図 19】

図19



【図 20】

図20



フロントページの続き

(72)発明者 小谷 博昭  
東京都青梅市今井2326番地 株式会社日立  
製作所デバイス開発センタ内

(72)発明者 野副 敦史  
東京都青梅市今井2326番地 株式会社日立  
製作所デバイス開発センタ内

(72)発明者 塩田 茂雅  
東京都小平市上水本町五丁目20番1号 株  
式会社日立製作所半導体事業部内

(72)発明者 片山 ゆかり  
神奈川県川崎市麻生区王禅寺1099番地株式  
会社日立製作所システム開発研究所内